On-line direct control design for nonlinear systems

On-line direct control design for nonlinear systems

Marko Tanaskovic [*] **Lorenzo Fagiano** [**] **Carlo Novara** [***]
Manfred Morari [*]

[*] *Automatic Control Laboratory, ETH Zurich, Physikstrasse 3, 8092 Zurich, Switzerland (e-mail: {tmarko,morari}@control.ee.ethz.ch)*
[**] *ABB Switzerland Ltd., Corporate Research, Segelhofstrasse 1K, 5405 Baden-Daettwil, Switzerland (e-mail: lorenzo.fagiano@ch.abb.com)*
[***] *Dipartimento di Automatica e Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy (e-mail: carlo.novara@polito.it)*

Abstract: An approach to design a feedback controller for nonlinear systems directly from experimental data is presented. Improving over a recently proposed technique, which employs exclusively a batch of experimental data collected in a preliminary experiment, here the control law is updated and refined during real-time operation, hence enabling an on-line learning capability. The theoretical properties of the described approach, in particular closed-loop stability and tracking accuracy, are discussed. Finally, the experimental results obtained with a water tank laboratory setup are presented.

Keywords: Data-driven control, nonlinear systems, direct control, dynamic inversion, stability

1. INTRODUCTION

Data-driven design techniques aim to derive a controller for a given dynamical system from experimental data, circumventing the need for a model based on physical first principles. In particular, *direct* data-driven approaches avoid the explicit derivation of even a black-box model of the system of interest, since they obtain directly a feedback controller from experimental data. In the literature concerned with direct data-driven control design, two main classes of approaches are found: on-line and off-line (batch). On-line techniques (see e.g. Hou and Jin (2013) and Helvoort et al. (2007)) can also be regarded to as direct adaptive control strategies, since the controller is modified with each new measurement obtained in closed loop. These techniques have the ability to exploit the data available during controller operation in order to improve its performance over time. However, due to the fact that the controller can change at any time, its behavior can be hard to predict and guaranteeing stability of these control schemes is challenging and often requires restrictive assumptions on the controlled system. In off-line approaches (see e.g. Sjöberg et al. (2003), Miskovic et al. (2007), Campi and Savaresi (2006) and Formentin et al. (2013)), the controller design is carried out using a batch of measurements collected in an initial experiment. The derived control law is then applied to the plant and it is not modified during operation. In most of the off-line techniques, stability and performance aspects are not explicitly considered in the design phase but rather evaluated via simulations or experiments before the controller becomes operational. In Novara et al. (2013), an off-line direct design approach that relies on nonlinear set-membership identification (see e.g. Milanese and Novara (2011)) has been proposed, which guarantees finite-gain stability of the closed loop system when the number of measurement points contained in the initial batch tends to infinity. The main disadvantage of off-line algorithms is that, unlike the on-line schemes, they do not exploit the additional measurements obtained during controller operation to improve the closed-loop performance. On the other hand, the behavior of the off-line designed controllers is more predictable.

In this paper we present a direct data-driven control design approach that aims to fuse on-line and off-line techniques, in order to retain the advantage of being able to improve the control performance over time while at the same time having a predictable closed loop behavior during operation. The proposed strategy enhances the off-line approach of Novara et al. (2013) by exploiting the theory of learning by projections (see e.g. Theodoridis et al. (2011) and the references therein) in order to recursively update the controller. Under reasonable assumptions on the initial batch of data, the approach guarantees stability of the closed loop system. Such guarantees are then retained during on-line learning, thanks to suitable robust constraints on parameters defining the feedback controller. The proposed approach is based on convex optimization and hence has moderate computational requirements. In addition to the new algorithm and its properties, we present the experimental results obtained with a water tank system, showing the advantage provided by the proposed learning scheme as compared to a purely off-line approach.

2. PROBLEM STATEMENT

We consider a discrete time nonlinear dynamical system with a single input described by the following state update equation:

$$x_{t+1} = g(x_t, u_t, e_t), \tag{1}$$

where $t \in \mathbb{Z}$ is the discrete time variable, $u_t \in U \subset \mathbb{R}$ is the control input, $x_t \in X \subset \mathbb{R}^{n_x}$ is the vector of states and $e_t \in \mathbb{R}^{n_e}$ is the vector of disturbance signals that accounts for both measurement and process disturbances. U and X are compact, possibly very large, sets bounding the system's input and state, respectively. We consider the following two assumptions on the disturbance e_t and function g:

Assumption 1. The value of e_t is bounded as

$$e_t \in B_\epsilon \doteq \{e_t : ||e_t|| \le \epsilon, \forall t \in \mathbb{Z}\},$$

for a finite value of $\epsilon > 0$. ■

Assumption 2. The function g is Lipschitz continuous with respect to u, i.e. for any $\tilde{x} \in X$ and $\tilde{e} \in B_\epsilon$, $g(\tilde{x}, \cdot, \tilde{e}) \in \mathcal{F}(\gamma_g, U)$, where

$$\mathcal{F}(\gamma_g, U) \doteq \begin{Bmatrix} g: ||g(0)|| < \infty, \\ ||g(u_1) - g(u_2)|| \leq \gamma_g ||u_1 - u_2||, \forall u_1, u_2 \in U \end{Bmatrix}.$$

■

Note that the notation $|| \cdot ||$ stands for a suitable vector norm chosen by the user (typically 2- or ∞-norm) and that the presented results hold for any norm.

It is assumed that the nonlinear function g is unknown, but a set $\mathcal{D}_N$ of N noise corrupted input and state measurements generated by the system (1) in an initial experiment (either in closed or in open loop) is available:

$$\mathcal{D}_N \doteq \{u_t, \omega_t\}_{t=-N}^{-1}, \, \omega_t \doteq (x_t, x_{t+1}).$$

In this paper, we consider the notion of finite gain stability (see e.g. Khalil (1996)):

Definition 2.1. A nonlinear system (possibly time varying) with input $u_t \in U$, state $x_t \in X$ and disturbance $e_t \in B_\epsilon$ as in (1) is *finite gain stable* from the input u to the state x if there exist finite and nonnegative constants λ and β such that:

$$||\mathbf{x}||_\infty \leq \lambda ||\mathbf{u}||_\infty + \beta, \forall \mathbf{u} \in \mathcal{U}, \forall \mathbf{e} \in \mathcal{B}_\epsilon,$$

where $\mathbf{x} = (x_1, x_2, \ldots)$, $\mathbf{u} = (u_1, u_2, \ldots)$, $\mathbf{e} = (e_1, e_2, \ldots)$, $||\mathbf{x}||_\infty \doteq \sup_k ||x_k||$ and $\mathcal{U}$ and $\mathcal{B}_\epsilon$ are the domains of the input and disturbance signals, respectively. ■

Based on this definition we make the following assumption about the nonlinear system (1):

Assumption 3. There exists a finite $\gamma > 0$ and a Lipschitz continuous function $f \in \mathcal{F}(\gamma, \mathbb{R}^{n_x} \times \mathbb{R}^{n_x})$ that makes the closed loop system with inputs $r_t \in \mathbb{R}^{n_x}$ and $v_t \in \mathbb{R}^{n_x}$:

$$x_{t+1} = g\left(x_t, f(x_t, r_{t+1}), e_t\right) + v_{t+1},$$

finite gain stable from the signals r and v to the state x. According to Definition 2.1, this means that there exist positive and finite constants λ_1, λ_2 and β such that

$$||\mathbf{x}||_\infty \leq \lambda_1 ||\mathbf{r}||_\infty + \lambda_2 ||\mathbf{v}||_\infty + \beta, \forall \mathbf{r} \in \mathcal{R}, \forall \mathbf{v} \in \mathcal{V}, \forall \mathbf{e} \in \mathcal{B}_\epsilon, \tag{2}$$

where $\mathbf{r} = (r_1, r_2, \ldots)$, $\mathbf{v} = (v_1, v_2, \ldots)$, and $\mathcal{R}$ and $\mathcal{V}$ are the domains of the signals r and v. ■

Assumption 3 states that the system (1) can be finite-gain stabilized by a Lipschitz continuous control function with respect to both the signal that is input to the controller (i.e. a reference signal) and a signal that is additive to the system state (i.e. an additive disturbance). Note that v_t and e_t are here separated for convenience, but v_t can actually be seen as a part of the disturbance e_t appearing in (1). Assumption 3 is thus very mild, since it basically requires that some controller exists which stabilizes the system (1).

We now state the control design problem addressed in this paper.

Problem 2.1. Based on the available noisy measurements, the goal is to initially design a feedback controller to track a desired reference signal $r_t \in R \subset X$ for $t > 0$, where R is a compact set and $||r_t|| \leq \bar{r}, \forall t > 0$, with $\bar{r} < \infty$ being the maximal magnitude of the reference signal. Moreover, once the controller is in operation, the algorithm should be capable of exploiting the incoming input and state measurements in order to improve the tracking performance of the controller while keeping the closed loop system finite gain stable. ■

3. ON-LINE DIRECT CONTROL DESIGN ALGORITHM

The main idea of the proposed control design approach is to learn from the available data an inverse of the system dynamics, and to use it as feedback controller. Following the definitions and notation introduced in Novara et al. (2013), we define the point-wise inversion error of a given controller f as:

$$IE(f, r, x, e) \doteq ||r - g(x, f(r, x), e)||, \tag{3}$$

and the global inversion error as:

$$GIE(f) \doteq ||IE(f, \cdot, \cdot, \cdot)||_\infty, \tag{4}$$

where $||\cdot||_\infty$ in (4) is the L_∞ function norm on $X \times R \times B_\epsilon$. Based on this, an optimal controller f^* is defined as:

$$f^* = \arg \min_{\mathcal{S} \cap \mathcal{F}_{\mathbb{R}^{n_x} \times \mathbb{R}^{n_x}}} GIE(f), \tag{5}$$

where $\mathcal{S}$ is the set of all functions f that satisfy Assumption 3 and $\mathcal{F}_{\mathbb{R}^{n_x} \times \mathbb{R}^{n_x}}$ denotes the set of all Lipschitz continuous functions on $\mathbb{R}^{n_x} \times \mathbb{R}^{n_x}$. We denote the Lipschitz constant of the function f^* by γ^* and the related constants λ_1, λ_2 and β of the closed loop system (obtained if the controller f^* would be used) by λ_1^*, λ_2^* and β^* (see (2)).

The optimal inverse function f^* is not known and it can not be exactly calculated, therefore it has to be approximated. Differently from Novara et al. (2013), where a time-invariant approximation of f^* is derived from the available training data $\mathcal{D}_N$, we present here an approach in which the optimal inverse function f^* is approximated by a time varying nonlinear function $\hat{f}_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \to \mathbb{R}$. This function is modified on-line based on both the available training data $\mathcal{D}_N$ and the incoming closed loop measurements in order to exploit all available information and obtain a better estimate of the optimal inverse function f^* at each time step and hence solve Problem 2.1.

The relation between the control input and state measurements can be written in the following way:

$$u_t = f^*(\omega_t) + d_t,$$

where d_t accounts for the influence of unmeasured disturbances and inversion errors. Since the signals x_t, u_t and e_t are bounded (i.e. they belong to compact sets) and the function g is Lipschitz continuous, it follows that the magnitude of the signal d_t has to be bounded, i.e. $d_t \in B_\delta \subset \mathbb{R}, \forall t \in \mathbb{Z}$, where $B_\delta \doteq \{d \in \mathbb{R} : |d| \leq \delta\}$, with δ being a positive constant. We will consider the bound δ to be known; in practice it can be estimated from the available training data (see e.g. Novara et al. (2013)). Based on this, and following the set membership identification approach, we can define the set of feasible inverse functions at time step t as:

$$\begin{aligned} FIFS_t &\doteq \bigcap_{k=-N,\ldots,t-1} H_k, \\ H_k &\doteq \{f \in \mathcal{F}_{\mathbb{R}^{n_x} \times \mathbb{R}^{n_x}} : |u_k - f(\omega_k)| \leq \delta\}. \end{aligned} \tag{6}$$

If Assumptions 1–3 hold, the optimal inverse function f^* has to belong to $FIFS_t$ (i.e. $f^* \in FIFS_t$) for all t. In the following, we will make use of this fact in order to bound the approximation error of the feedback controller $\hat{f}_t \approx f^*$ and guarantee closed loop stability.

We parametrize the approximate controller with kernel functions, which means that at each time step the function $\hat{f}_t$ is given by:

$$\hat{f}_t(\omega) = a_t^T K(\omega, W_t),$$

where $a_t \in \mathbb{R}^{L_t}$ is the vector of weights, and $K(\omega, W_t) = [\kappa(\omega, \tilde{\omega}_1), \ldots, \kappa(\omega, \tilde{\omega}_{L_t})]^T$ is a vector formed by stacking the values of the kernel functions $\kappa(\cdot, \tilde{\omega}_i) : \mathbb{R}^{2n_x} \to \mathbb{R}, i =$

$1, \ldots, L_t$ contained in a dictionary of functions that is uniquely determined by the set of L_t kernel function centers $W_t = \{\tilde{\omega}_1, \ldots, \tilde{\omega}_{L_t}\}$. The set W_t which determines the kernel function dictionary and the vector of weights a_t are updated at each time step based on the incoming input and state measurements. In the following subsections, we first introduce the inequality that we use to bound the approximation error of the controller $\hat{f}_t$ at each time step and to impose finite gain stability, then we discuss in detail the mechanisms of introducing new kernel functions to the dictionary and the algorithm for updating the vector of weights. Finally, we summarize the proposed design method and discuss its computational and memory requirements.

3.1 Inequality to enforce closed loop stability

In order to have finite gain stability of the closed loop, we require the function $\hat{f}_t$, to satisfy the following robust inequality at each time step t:

$$\begin{aligned} &|\hat{f}_t(\omega_t^+) - f^*(\omega_t^+)| \leq \gamma_\Delta ||x_t|| + \sigma, \\ &\forall f^* \in \mathcal{F}(\gamma^*, \mathbb{R}^{n_x} \times \mathbb{R}^{n_x}) \cap FIFS_0, \, \forall t, \end{aligned} \tag{7}$$

where $\omega_t^+ = [x_t, r_{t+1}]^T$ if $t \geq 0$ and $\omega_t^+ = [x_t, x_{t+1}]^T$ otherwise, and $\gamma_\Delta, \sigma \in \mathbb{R}, \gamma_\Delta, \sigma > 0$, are design parameters. Guidelines on how these tuning parameters should be selected in order to guarantee finite gain stability of the closed loop are provided in Section 4. For $t \geq 0$, the inequality (7) requires the absolute difference between the control input calculated by $\hat{f}_t$ at time step t, i.e. $u_t = \hat{f}_t(\omega_t^+)$, and the one given by the optimal inverse function f^*, to be not larger than a term that depends linearly on the norm of the current state. Since the actual ideal inverse function f^* is not known, we require this inequality to be satisfied robustly for all functions in $FIFS_0$ that have the Lipschitz constant equal to γ^*. Note that in order to impose the constraint in (7), the Lipschitz constant γ^* of f^* needs to be known. The value of γ^* can be estimated in an off-line procedure from the collected data $\mathcal{D}_N$ by using the method presented in Novara et al. (2013).

In order to obtain a computationally tractable version of inequality (7), we can use the method proposed in Milanese and Novara (2011) in order to calculate an upper and a lower bound on the optimal inverse function $f^* \in FIFS_0$ that we denote by $\overline{f}$ and $\underline{f}$ respectively:

$$\begin{aligned} \overline{f}(\omega) &= \min_{k=-N,\ldots,-1} \left(u_k + \delta + \gamma^* ||\omega - \omega_k||_\infty\right) \\ \underline{f}(\omega) &= \max_{k=-N,\ldots,-1} \left(u_k - \delta - \gamma^* ||\omega - \omega_k||_\infty\right). \end{aligned} \tag{8}$$

It then follows that the inequality (7) can be satisfied by enforcing the following two inequalities:

$$-\gamma_\Delta ||x_t|| - \sigma + \overline{f}(\omega_t^+) \leq \hat{f}_t(\omega_t^+) \leq \gamma_\Delta ||x_t|| + \sigma + \underline{f}(\omega_t^+) \tag{9}$$

Note that instead of all $f^* \in \mathcal{F}(\gamma^*, \mathbb{R}^{n_x} \times \mathbb{R}^{n_x}) \cap FIFS_0$, the robust constraint (7) could be enforced for all $f^* \in \mathcal{F}(\gamma^*, \mathbb{R}^{n_x} \times \mathbb{R}^{n_x}) \cap FIFS_t$, which would be a less conservative condition since $FIFS_t \subseteq FIFS_0$ (see e.g. (6)). However, practical enforcement of such a condition would be quite difficult as the number of points over which the min and max in (8) need to be evaluated would increase with each time step.

3.2 Updating the dictionary of kernel functions

Kernel functions are widely used by the machine learning community for parameterizations in nonlinear approximation and learning tasks (see e.g. Schölkopf and Smola (2001)). Any Lipschitz continuous nonlinear function can be well approximated by a dictionary of functions that are centered at the points at which the function is evaluated. Since we aim at having a recursive scheme for updating the controller $\hat{f}_t$ at each time step, we allow for the dictionary to grow and incorporate new elements as new input and state measurements arrive. However, in order to prevent an unlimited growth of the dictionary size over time, we chose to add an element ω to W_t only if the kernel function centered at ω is sufficiently different from all the kernel functions centered at points already in W_t. To this end we use the coherence factor (see e.g. Richard et al. (2009)) of the incoming data point ω with respect to the set W_t:

$$\mu(\omega, W_t) = \max_{i=1,\ldots,L_t} \frac{|\kappa(\omega, \tilde{\omega}_i)|}{\sqrt{|\kappa(\omega, \omega)|}\sqrt{|\kappa(\tilde{\omega}_i, \tilde{\omega}_i)|}}. \tag{10}$$

Note that $\mu(\omega, W_t) \in (0, 1]$, and that the larger the coherence value in (10), the more similar is the kernel function centered at ω to some kernel function already in the dictionary. Therefore, for each incoming measurement point ω, we compare its coherence factor (10) with a threshold $\overline{\mu} \in (0, 1)$ and only add it to W_t if $\mu(\omega, W_t) \leq \overline{\mu}$. The threshold $\overline{\mu}$ is a tuning parameter that should be chosen by the control designer and it determines the overall size and density of the function centers forming the dictionary.

An important question that should be addressed is whether the size of the dictionary obtained by using the described update rule remains limited over time. To this end, we recall an important property of the coherence measure that was demonstrated in Richard et al. (2009).

Lemma 3.1. (Proposition 2 in Richard et al. (2009)) Let $\mathcal{W}$ be a compact set. Then for any $\overline{\mu} \in (0, 1)$, the dictionary obtained by adding a kernel function centered at $\omega_t \in \mathcal{W}$ to it when $\mu(\omega_t, W_t) \leq \overline{\mu}$ has a finite number of elements for any sequence $\{w_t\}_{t=-N}^{\infty}$. ■

Since the state of the system belongs to a compact set X, i.e. $x_t \in X, \forall t$, according to Lemma 3.1, the size of the dictionary remains bounded over time.

3.3 Updating the vector of weights

In order to describe the algorithm for the iterative update of the vector $a_t \in \mathbb{R}^{L_t}$, we note that the input and the state measurement u_j and ω_j and the dictionary at time step t define the following set in which the vector a_t should lie for the function f_t to belong to the set H_j defined in (6), i.e. $\hat{f}_t \in H_j$:

$$S_{jt} \doteq \{a \in \mathbb{R}^{L_t} : |a^T K(\omega_j, W_t) - u_j| \leq \delta\}.$$

The set S_{jt} is a strip (hyperslab) in $\mathbb{R}^{L_t}$. We further define the projection of a point in $\mathbb{R}^{L_t}$ onto the strip S_{jt} as:

$$P_{jt}(a) \doteq \min_{\hat{a} \in S_{jt}} ||a - \hat{a}||_2. \tag{11}$$

Note that the solution of the convex optimization problem (11) can be explicitly derived (see e.g. Theodoridis et al. (2011)) and it has the following form:

$$P_{jt}(a) = a + \begin{cases} \dfrac{u_j - \delta - a^T K(\omega_j, W_t)}{||K(\omega_j, W_t)||_2^2} K(\omega_j, W_t) & \text{if } u_j - \delta > a^T K(\omega_j, W_t) \\ 0 & \text{if } |a^T K(\omega_j, W_t) - u_j| \leq \delta \\ \dfrac{u_j + \delta - a^T K(\omega_j, W_t)}{||K(\omega_j, W_t)||_2^2} K(\omega_j, W_t) & \text{if } u_j + \delta < a^T K(\omega_j, W_t). \end{cases} \tag{12}$$

In addition, we define the hyperslab to which the vector a_t should belong to in order to satisfy the stability constraint (9) as:

$$S_t^+ \doteq \left\{ \begin{array}{r} a \in \mathbb{R}^{L_t} : -\gamma_\Delta ||x_t|| - \sigma + \overline{f}(\omega_t^+) \leq a^T K(\omega_t^+, W_t) \\ a^T K(\omega_t^+, W_t) \leq \gamma_\Delta ||x_t|| + \sigma + \underline{f}(\omega_t^+) \end{array} \right\}.$$

Moreover, analogously to the definition of $P_{jt}(a)$ in (12), we denote the projection of any point $a \in \mathbb{R}^{L_t}$ onto the hyperslab S_t^+ by $P_t^+(a)$. Note that this projection can also be calculated explicitly by using a formula similar to (12). In the recursive algorithm that we propose, a_t is calculated from a_{t-1}. However, since the size of the dictionary can expand from time step $t-1$ to time step t, in general it will hold that $a_{t-1} \in \mathbb{R}^{L_{t-1}}$ and $a_t \in \mathbb{R}^{L_t}$ with $L_{t-1} \leq L_t$. Therefore, we define $a_{t-1}^+ \in \mathbb{R}^{L_t}$ as the extension of the vector a_{t-1} obtained by appending the appropriate number of zeros to a_{t-1} in order to get a vector of dimension L_t:

$$a_{t-1}^+ = [a_{t-1}^T, \underbrace{0, \ldots, 0}_{L_t - L_{t-1}}]^T. \quad (13)$$

In order to update the vector of weights at each time step, we use the same idea exploited by the projection learning algorithms, that by repeatedly projecting a point onto convex sets it eventually ends up in their intersection. To this end let us define the set of indexes $J_t = \{\max\{-N, t-q\}, \ldots, t-1\}$ that will be used to denote the last q collected state and input measurements, where $q \in \mathbb{N}, q > 0$ is a tuning parameter, and let $I_t = \{j \in J_t : a_{t-1}^+ \notin S_{jt}\}$ be the set of indexes in J_k that have the property that the weighting vector a_{t-1}^+ is not in the measurement strip associated to those indexes. Based on this, we can write the equation for calculating the weighting vector a_t from a_{t-1}^+ as:

$$a_t = P_t^+ \left(a_{t-1}^+ + \sum_{j \in I_t} \frac{1}{card(I_t)} \left(P_{jt}(a_{t-1}^+) - a_{t-1}^+ \right) \right), \quad (14)$$

where $card(I_t)$ denotes the number of elements in I_t.

According to (14), the vector of weights a_t is calculated by first finding the convex combination of the projections of the vector a_{t-1}^+ to the hyperslabs defined by the latest q input and state measurements. This point is then projected onto the hyperslab S_t^+. Hence, the update rule (14) steers the vector of weights a_t towards the intersection of the hyperslabs defined by the incoming input and state measurements, while always enforcing the satisfaction of the constraint (9).

In order to state the convergence property of the update rule (14), we note that from Lemma 3.1 it follows that there exists some finite time step $\overline{t} < \infty$ such that $L_t = \overline{L}, \forall t \geq \overline{t}$ and that $\forall t_1, t_2 \geq \overline{t}$ it holds that $S_{jt_1} = S_{jt_2} \subset \mathbb{R}^{\overline{L}}$ and therefore for $t \geq \overline{t}$, we can denote S_{jt} just by S_j. Based on this, we state the main property of the update rule(14).

Lemma 3.2. (Slight modification of Theorem 4.2 in Slavakis and Yamada (2013)) Let $\Omega \doteq \bigcap_{t \geq t_0} \left(S_t^+ \bigcap \left(\bigcap_{j \in I_t} S_j \right) \right) \neq \emptyset$ for some finite $t_0 \in \mathbb{N}, \overline{t} \leq t_0 < \infty$. Then the recursive update rule (14) is guaranteed to bring the point $a_t \in \mathbb{R}^{\overline{L}}$ closer to the set Ω with each time step $t \geq t_0$, i.e. $\min_{a \in \Omega} ||a_t - a||_2 \leq \min_{a \in \Omega} ||a_{t-1} - a||_2, \forall t \geq t_0$ and in the limit it holds that:

$$\lim_{t \to \infty} a_t \in \left[\left(\bigcup_{n=\overline{t}}^{\infty} \bigcap_{j \geq n} S_j \right) \bigcap \left(\bigcup_{n=\overline{t}}^{\infty} \bigcap_{t \geq n} S_t^+ \right) \right].$$

■

Lemma 3.2 states that if the set Ω is nonempty, then in the limit the vector of weights a_t is guaranteed to belong to the intersection of all but finitely many hyperslabs S_t^+ and $S_j, j \in I_t, t \geq \overline{t}$. Hence there exists some finite time step $\hat{t} \in \mathbb{N}, \overline{t} \leq \hat{t} < \infty$ such that $\lim_{t \to \infty} f_t \in \bigcap_{k \geq \hat{t}} H_k$. Therefore, as the number of collected measurement points tends to infinity, the approximate controller is guaranteed to end up in the intersection of all, but finitely many sets H_k (see (6)) defined by the collected measurements.

3.4 Summary of the proposed design algorithm

The described procedures to update the dictionary and the weighting vector can be merged into an on-line scheme to design feedback control law on the basis of both the initial training data $\mathcal{D}_N$ and the additional measurements obtained in closed loop. In Algorithm 3.1 we state such an approach.

Algorithm 3.1. Feedback control algorithm based on the on-line control design scheme

1) Collect the state measurement x_t;
2) Form W_t from W_{t-1} by adding ω_{t-1} if $\mu(\omega_{t-1}, W_{t-1}) \leq \overline{\mu}$ and ω_t^+ if $\mu(\omega_t^+, W_{t-1}) \leq \overline{\mu}$. Form the vector a_{t-1}^+ according to (13);
3) Calculate a_t by using (14);
4) If $t \geq 0$, calculate $u_t = a_t^T K(\omega_t^+, W_t)$ and apply it to the plant;
5) Set $t = t + 1$ and go to 1).

For $t \geq 0$ Algorithm 3.1 is both a controller and a design algorithm and can therefore be seen as an adaptive controller; for $t < 0$ it only acts as a design algorithm. The proposed scheme has moderate computational requirements as it does not require solving complex mathematical problems and since many operations needed for updating the vector of weights in (14) can be parallelized. In addition, the number of kernel functions in the dictionary remains bounded over time, as demonstrated in Section 3.2. However, in order to evaluate the bounds (8), the training data $\mathcal{D}_N$ need to be stored in memory and available during the controller runtime.

4. PROPERTIES OF THE PROPOSED DIRECT DESIGN ALGORITHM

In order to state the main property of the proposed algorithm, we make the following assumption on the selection of the tuning parameter γ_Δ.

Assumption 4. The tuning parameter $\gamma_\Delta \in \left(0, \frac{1}{\gamma_g \lambda_2^*}\right)$. ■

Based on this assumption, we define the maximal achievable state amplitude as:

$$\overline{x} \doteq \frac{\lambda_1^*}{1 - \gamma_g \lambda_2^* \gamma_\Delta} \overline{r} + \frac{\gamma_g \lambda_2^*}{1 - \gamma_g \lambda_2^* \gamma_\Delta} \sigma + \frac{\beta^*}{1 - \gamma_g \lambda_2^* \gamma_\Delta}, \quad (15)$$

and the sets $B_{\overline{x}}$ and $B_{\overline{xr}}$ as:

$$\begin{aligned} B_{\overline{x}} &\doteq \{x \in \mathbb{R}^{n_x} : ||x|| \leq \overline{x}\}, \\ B_{\overline{xr}} &\doteq \{\omega \in \mathbb{R}^{n_x} \times R : \omega = (x, r), \forall x \in B_{\overline{x}}, r \in R\}. \end{aligned} \quad (16)$$

We denote the maximal possible difference between the upper and the lower bound on the value of the function

f^* on the set $B_{\overline{xr}}$ that is calculated based on the collected measurements $\mathcal{D}_N$ by D_0:

$$D_0 \doteq \sup_{\omega \in B_{\overline{xr}}} \left(\overline{f}(\omega) - \underline{f}(\omega)\right). \tag{17}$$

D_0 is called the diameter of information and in the set membership framework is used to measure the uncertainty associated with the identification/design process. Based on these definitions, we make the following assumption.

Assumption 5. The selected value of σ and the training data $\mathcal{D}_N$ are such that $\sigma \geq \frac{D_0}{2}$. ■

In order to verify whether Assumptions 4 and 5 hold, the values of γ_g, λ_1^*, λ_2^* and β^* should be known. The Lipschitz constant γ_g can be estimated from the available training data (see e.g. Novara et al. (2013)). The values of λ_1^*, λ_2^* and β^* can not be estimated based on the available data and they have to be guessed. Since these parameters are related to the performance of the optimal inverse controller f^* that should typically result in small tracking error (i.e. the state of the corresponding closed loop system should be close to the desired reference signal), a reasonable guess for λ_1^* and λ_2^* is a value slightly greater than 1 and for β^* a value close to 0. In addition, efficient numerical algorithms for estimating D_0 from the training data $\mathcal{D}_N$ are available (see e.g. Milanese and Novara (2007)).

Assumption 4 can be easily satisfied by selecting a suitable value for γ_Δ. On the other hand, from (15), (16) and (17) it follows that the diameter of information D_0 depends on the selected value of σ, which could make the selection of σ that satisfies Assumption 5 challenging. However, the value of D_0 can only decrease as the size of the training data $\mathcal{D}_N$ increases. Therefore, if the measurements generated in the initial experiment are informative enough, then for a fixed value of σ, a finite number N of training data $\mathcal{D}_N$ should be collected such that the condition of Assumption 5 is satisfied (see e.g. Milanese and Novara (2011) for details).

We also make the following technical assumption that relates the boxes $B_{\overline{x}}$ and $B_{\overline{xr}}$ and the sets U and X.

Assumption 6. $B_{\overline{x}} \subseteq X$. Moreover, $\forall \omega \in B_{\overline{xr}}, \forall \Delta u \in [-\gamma_\Delta \overline{x} - \sigma, \gamma_\Delta \overline{x} + \sigma], f^*(\omega) + \Delta u \in U$. ■

Assumption 6 requires the compact sets X and U, in which the state and the input of the plant (1) evolve, to be sufficiently large such that they include the box $B_{\overline{x}}$ and all possible control inputs evaluated on the basis of state measurement and reference value pars in $B_{\overline{xr}}$ respectively.

We now have all the ingredients to state the theorem on the finite gain stability of the closed loop.

Theorem 4.1. Let the Assumptions 1–5 hold and let $S_0^+ \neq \emptyset$ and $x_0 \in B_{\overline{x}}$. Then for a reference signal $r_t \in R, ||r_t|| \leq \overline{r}, \forall t > 0$, it holds that $S_t^+ \neq \emptyset, \forall t \geq 0$ and the closed loop system is finite gain stable from the reference r_t to the state x_t when Algorithm 3.1 is used.

Proof 4.1. We will prove the theorem by induction. But first, we note that the closed loop system obtained by using the approximate controller $\hat{f}_t$ can be represented as:

$$\begin{aligned} x_{t+1} &= g\left(x_t, \hat{f}_t(x_t, r_{t+1}), e_t\right) = g\left(x_t, f^*(x_t, r_{t+1}), e_t\right) + v_{t+1}, \\ v_{t+1} &= g\left(x_t, \hat{f}_t(x_t, r_{t+1}), e_t\right) - g\left(x_t, f^*(x_t, r_{t+1}), e_t\right). \end{aligned} \tag{18}$$

From Assumption 3 and the definition of the optimal inverse controller f^* in (5), it holds that:

$$||\mathbf{x}||_\infty \leq \lambda_1^* ||\mathbf{r}||_\infty + \lambda_2^* ||\mathbf{v}||_\infty + \beta^*. \tag{19}$$

Moreover, we note that from Assumptions 2 and 6, it follows that:

$$||v_{t+1}|| \leq \gamma_g |\hat{f}_t(x_t, r_{t+1}) - f^*(x_t, r_{t+1})|, \forall (x_t, r_{t+1}) \in B_{\overline{xr}}. \tag{20}$$

We now employ the inductive argument to show that if $S_0^+ \neq \emptyset$ and $x_0 \in B_{\overline{x}}$, then $S_t^+ \neq \emptyset$ and $x_t \in B_{\overline{x}}, \forall t \geq 0$. The condition is satisfied for $t = 0$ by the Theorem assumption. Let us assume, for the sake of inductive argument, that $S_k^+ \neq \emptyset$ and $x_k \in B_{\overline{x}}, \forall k \in [0, t-1]$. From this assumption and the way the weighting vector a_t is updated in (14), it follows that $a_k \in S_k^+$, $\forall k \in [0, t-1]$, which means that the robust condition (7) is satisfied for all $k \in [0, t-1]$ and therefore, due to Assumptions 1 and 2, it holds that:

$$|\hat{f}_k(x_k, r_{k+1}) - f^*(x_k, r_{k+1})| \leq \gamma_\Delta ||x_k|| + \sigma, \forall k \in [0, t-1]. \tag{21}$$

From (20) and (21) it then follows that:

$$||v_{k+1}|| \leq \gamma_g \gamma_\Delta ||x_k|| + \gamma_g \sigma, \forall k \in [0, t-1]. \tag{22}$$

From Assumption 4, it follows that $\gamma_g \lambda_2^* \gamma_\Delta < 1$. From this fact and from (19) and (22) it further follows that:

$$||\mathbf{x_t}||_\infty \leq \frac{\lambda_1^*}{1 - \gamma_g \lambda_2^* \gamma_\Delta} ||\mathbf{r_t}||_\infty + \frac{\gamma_g \lambda_2^*}{1 - \gamma_g \lambda_2^* \gamma_\Delta} \sigma + \frac{\beta^*}{1 - \gamma_g \lambda_2^* \gamma_\Delta}, \tag{23}$$

where $\mathbf{x_t} = (x_1, \ldots, x_t)$ and $\mathbf{r_t} = (r_1, \ldots, r_t)$. From the fact that $||\mathbf{r_t}||_\infty \leq \overline{r}$ and from (15), it then follows that $||x_t|| \leq \overline{x}$, i.e. $x_t \in B_{\overline{x}}$. Therefore, $\omega_t^+ \in B_{\overline{xr}}$ and from (17), it holds that $\overline{f}(\omega_t^+) - \underline{f}(\omega_t^+) \leq D_0$. From Assumption 5, it then follows that:

$$-\gamma_\Delta ||x_t|| - \sigma + \overline{f}(\omega_t^+) \leq \gamma_\Delta ||x_t|| + \sigma + \underline{f}(\omega_t^+),$$

which implies that $S_t^+ \neq \emptyset$. Repeating this inductive argumentation for all $t \geq 0$, it follows that $S_t^+ \neq \emptyset, \forall t \geq 0$. In addition, (23) will hold for all $t \geq 0$ which implies that the closed loop system is finite gain stable from the reference to the state (see e.g. Definition 2.1). ■

A direct consequence of Theorem 4.1 and its proof is that the tracking error remains bounded $\forall t \geq 0$. Namely, from (3), (18) and (20) it follows that $\forall t \geq 0$:

$$||r_t - x_t|| \leq IE(f^*, r_t, x_t, e_t) + \gamma_g |\hat{f}_t(x_{t-1}, r_t) - f^*(x_{t-1}, r_t)|.$$

Therefore, it is reasonable to expect the tracking error to decrease as more data become available and the approximation accuracy of the function $\hat{f}_t$ becomes better. Moreover, from the fact that the approximation error of $\hat{f}_t$ is kept bounded $\forall t > 0$, it follows that the tracking error always remains upper bounded by a function that linearly depends on the norm of the state:

$$||r_t - x_t|| \leq IE(f^*, r_t, x_t, e_t) + \gamma_g \gamma_\Delta ||x_{t-1}|| + \gamma_g \sigma, \forall t \geq 0. \tag{24}$$

Therefore, despite the fact that the controller $\hat{f}_t$ can change at each time step during operation, its behavior is well determined as an upper bound on the tracking error can be theoretically derived.

Note that a compromise has to be made in the selection of the tuning parameters γ_Δ and σ. Namely, taking small values for γ_Δ and σ leads to a tighter upper bound on the tracking error in (24). However, taking γ_Δ too small may lead to the set Ω (see e.g. Lemma 3.2) being empty and taking σ too small may require a very large number of initial training data in order to satisfy Assumption 5.

5. EXPERIMENTAL RESULTS

The performance of the proposed control design algorithm was tested experimentally on a water tank system. The system consists of a round water tank that has a water inlet at the top and a small opening at its bottom through which the water leaks out of the tank. Water is injected into the tank by a pump whose voltage v is the control input. The pump has a nonlinear characteristic. Water level in the tank, that we denote by h, is the controlled variable and can be measured by a pressure sensor located at the bottom of the tank. Hence, with reference to the notation used in the previous sections, we have $x = h$ and $u = v$. A sketch of the described experimental setup and a picture of it are shown in Fig. 1.

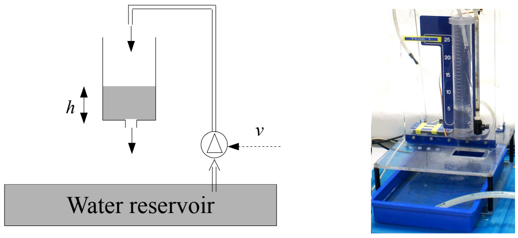

Fig. 1. Exp. setup schematic (left) and photo (right).

A sampling time of 3 seconds was selected for controlling the system. In order to implement the proposed design scheme, we generated a 4000 sample long training data $\mathcal{D}_N$. Based on the recorded data, by using the procedure described in Novara et al. (2013), the noise bound δ and the Lipschitz constants γ_g and γ^* were estimated to be 0.15, 1.25 and 3.6 respectively. We selected the values of λ_1^* and λ_2^* to be 1.12 and the value of β^* to be 0, and based on this we chose the value of γ_Δ to be 0.15. In addition, from the available training data $\mathcal{D}_N$, by using the method proposed in Milanese and Novara (2007), we calculated the value of D_0 in (17) over the set $B_{\overline{xr}}$ defined by $\overline{x} = 18$ and $\overline{r} = 9$ to be $D_0 = 5.8$. We used the Gauss kernel functions (see e.g. Schölkopf and Smola (2001)) to parametrize the controller. All relevant tuning parameters used by the algorithm are listed in Table 1. Note that the parameters were selected such that Assumptions 4 and 5 are valid.

Table 1. Tuning parameters.

δ	γ^*	γ_Δ	β	$\overline{\mu}$	q
0.15	3.6	0.15	2.9	0.82	20

In order to further illustrate the advantage of using the proposed on-line scheme, we compared the performance of the Algorithm 3.1 for the case when the updating of the controller $\hat{f}_t$ is stopped before the controller is put into operation (at time step $t = 0$) and the case when the updating is also done during controller operation. The reference tracking performance for the two cases is compared in Fig. 2. Although both controllers exhibit good tracking performance, the performance of the controller that is updated on-line improves over time as more data becomes available. The average tracking error for the controller that is updated on-line is 0.13 cm compared to 0.35 cm for the controller that is designed on the basis of initial training data only. This illustrates the main advantage of the proposed scheme, which is the ability to exploit the incoming measurements obtained during controller operation and have a predictable behavior at the same time.

6. CONCLUSION

In this paper, we have proposed a novel on-line direct inversion based control design method. The algorithm

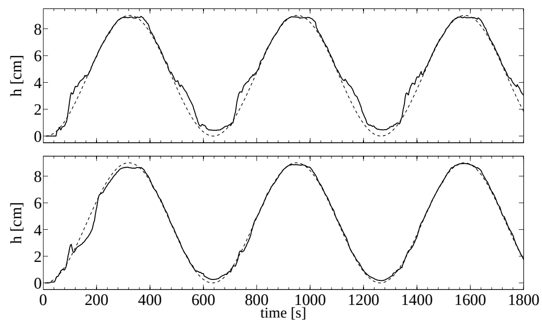

Fig. 2. Measured tank water level (tick lines) and the desired reference (dashed line) for the case when there is no on-line update (upper plot) and when the controller is updated on-line (lower plot)

exploits the results from the set membership theory for nonlinear systems and the theory of learning by projections in order to bridge the gap between the existing on-line and off-line direct control design schemes. Properties of the proposed algorithm have been analyzed theoretically and its advantages were demonstrated experimentally on a water tank system.

REFERENCES

Campi, M.C. and Savaresi, S.M. (2006). Direct nonlinear control design: the virtual reference feedback tuning (VRFT) approach. *IEEE Transactions on Automatic Control*, 51, 14–27.

Formentin, S., De Filippi, P., Corno, M., Tanelli, M., and Savaresi, S.M. (2013). Data-driven design of braking control systems. *IEEE Transactions on Control Systems Technology*, 21, 186–193.

Helvoort, J.V., de Jager, B., and Steinbuch, M. (2007). Direct data-driven recursive controller unfalsification with analytic update. *Automatica*, 43, 2034–2046.

Hou, Z. and Jin, S. (2013). *Model free adaptive control: theory and applications*. CRC Press.

Khalil, H.K. (1996). *Nonlinear systems*. Prentice Hall.

Milanese, M. and Novara, C. (2007). Computation of local radius of information in SM-IBC identification of nonlinear systems. *Journal of Complexity*, 23, 937–951.

Milanese, M. and Novara, C. (2011). Unified set membership theory for identification, prediction and filtering of nonlinear systems. *Automatica*, 47, 2141–2151.

Miskovic, L., Karimi, A., Bonvin, D., and Gevers, M. (2007). Corelation-based tuning of decoupling multivariable controllers. *Automatica*, 43, 1482–1494.

Novara, C., Fagiano, L., and Milanese, M. (2013). Direct feedback control design for nonlinear systems. *Automatica*, 49, 849–860.

Richard, C., Bermudez, J.C.M., and Honeine, P. (2009). Online prediction of time series data with kernels. *IEEE Transactions on Signal Processing*, 57, 1058–1067.

Schölkopf, B. and Smola, A.J. (2001). *Learning with kernels*. MIT Press, Cambridge MA.

Sjöberg, J., Bruyne, F.D., Agarwal, M., Anderson, B.D.O., Gevers, M., Kraus, F.J., and Linard, N. (2003). Iterative controller optimization for nonlinear systems. *Control Engineering Practice*, 11, 1079–1086.

Slavakis, K. and Yamada, I. (2013). The adaptive projected subgradient method constrained by families of quasi-nonexpansive mappings and its application to online learning. *SIAM Journal of Optimization*, 23, 126–152.

Theodoridis, S., Slavakis, K., and Yamada, I. (2011). Adaptive learning in a world of projections. *IEEE Signal Processing Magazine*, 28, 97–123.